(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 938 207 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
25.08.1999 Bulletin 1999/34

(51) Int Cl.[6]: **H04L 1/12**, H04L 1/18

(21) Application number: **99301300.2**

(22) Date of filing: **23.02.1999**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **23.02.1998 US 75501 P**

(71) Applicant: **LUCENT TECHNOLOGIES INC.**
Murray Hill, New Jersey 07974-0636 (US)

(72) Inventors:
- **Balachandran, Krishna**
  Middletown, New Jersey 07748 (US)
- **Ejzak, Richard Paul**
  Wheaton, Illinois 60187 (US)
- **Nanda, Sanjiv**
  Clarksburg, New Jersey 08510 (US)

(74) Representative:
Buckley, Christopher Simon Thirsk et al
Lucent Technologies (UK) Ltd,
5 Mornington Road
Woodford Green, Essex IG8 0TU (GB)

(54) System and method for incremental redundancy transmission in a communication system

(57) A system and method for incremental redundancy transmission in a communication system. A time slot is provided having at least one sub-slot with a fixed size, and a data block sized to fit in the sub-slot, a header having a one data block sequence number in the header for the time slot. A parity block is sized smaller than the data block such that the parity block and the data block sequence number fit within the sub-slot. The data block and parity block are trnasmitted in the sub-slot within the time slot. In the header for the time slot the number of data blocks and parity blocks transmitted are identified.

EP 0 938 207 A2

Printed by Jouve, 75001 PARIS (FR)

BNSDOCID: <EP___0938207A2_I_>

Description

Cross Reference To Related Application

[0001] This application claims the benefit of U.S. Provisional Application, Serial No. 60/075,501, filed February 23, 1998, and entitled System and Method for Incremental Redundancy Transmission in a Communication System.

Background Of The Invention

[0002] This invention relates to incremental redundancy transmission in a communication system, and more particularly, time slotted communication systems.

[0003] Link layer recovery protocols are used for error and loss recovery in data communication systems. Link layer recovery is especially crucial for wireless communications due to the particularly harsh loss and error characteristics of the link.

[0004] When a lost data frame is retransmitted, the receiver may combine the multiple received copies of the frame to increase the likelihood of correct decoding. Alternately, the transmitter may transmit additional parity information instead of retransmitting another copy of the lost frame. These methods of incremental redundancy transmission and soft combining are well known in the art.

[0005] Despite having performance potential, these methods have not found an application in practical systems. To enable efficient transmission, (that is, high rate coding), it is required that the protocol permits the transmission of proportionally small amounts of incremental parity information.

[0006] The problems associated with the efficient transmission were addressed in a paper, "A Proposal for IS- 136+ Data Services" by Robert Van Nobelen and Nambirajan Seshadri of AT&T Laboratories and Krishna Balachandran, Richard Ejzak and Sanjiv Nanda of Bell Labs, Lucent Technologies Inc. attached herewith as Appendix I, the disclosure of which is hereby incorporated by reference. The paper proposed that the problem be handled by breaking up an uncoded data frame into D blocks, and introducing P parity blocks. Initially, all data blocks that constitute a frame are transmitted. If the receiver is unable to reconstruct, the transmitted frame parity blocks are transmitted one at a time until the receiver is able to recover the frame. If all P parity frames are transmitted, the corresponding coding rate is $\frac{D}{D+P}$.

[0007] Practical implementation of the proposed procedure requires transmission of frame and block sequence numbers for each block. In addition, variable sized blocks are used at different modulation formats (e.g., QPSK, 8 PSK and 16 PSK).

[0008] The present invention is directed to overcoming, or at least reducing, the effects of one or more of the problems set forth above.

Summary Of The Invention

[0009] In accordance with the present invention, there is provided a system and method for designing an incremental redundancy, adaptive modulation data communication system using a time slotted communication channel. The method performs the steps of establishing a time slot having at least one sub-slot with a fixed size and establishes a data block sized to fit in the sub-slot. A header is established having a data block sequence number in the header for the time slot. A parity block is also established which is sized smaller than the data block such that the parity block and data block sequence number fit within the sub-slot. The data block and parity block are transmitted in a sub-slot within the time slot. The method further performs the step of identifying, in the header for the time slot, the number of data blocks and parity blocks transmitted.

[0010] Stated generally, the present invention encompasses an incremental redundancy transmission communication system. The system comprises a time slot signal generator that creates a time slot signal having at least one sub-slot with a fixed size and a data block signal generator that creates a data block signal sized to fit in the time slot. A header adder that generates a header having a data block signal sequence number in the header for the time slot signal is also provided. A receiver is also provided that identifies in the header for the time slot signal, a number of data block signals consecutively transmitted.

[0011] The present invention further encompasses a method for incremental redundancy transmission in a communication system that uses radio (or retransmission) protocol link frames. The method comprises the step of creating a time slot signal having at least one sub-slot with a fixed size, creating a data block signal sized to fit in the sub-slot, establishing a header having a data block signal sequence number in the header for the time slot signal and identifying in the header for the time slot signal, the number of data block signals consecutively transmitted. The steps of providing, in a header of the time slot signal, a data block sequence number for a first data block signal in a sequence of data block signals and creating a parity block signal smaller than the data block signal such that the parity block signal sequence number fits within the sub-slot signal are also provided.

BNSDOCID: <EP___0938207A2_I_>

[0012] These and other features and advantages of the present invention will become apparent from the following detailed description, the accompanying drawings and the appended claims.

Brief Description Of The Drawings

[0013]

Fig. 1 shows a block diagram describing operation on a packet data channel in accordance with the invention;

Fig. 2 shows a block diagram of the transmitter operation for Mode 0 and shows the blocks involved in conversion of Layer 3 frames into Layer 1 frames;

Fig. 3 shows a block diagram of the transmitter operation for Mode 1 and shows the blocks involved in conversion of Layer 3 frames into Layer 1 frames;

Fig. 4 shows a block diagram describing Mode 0 with a base station to mobile communication link and feedback interactions between the uplink and downlink;

Fig. 5 illustrates the format for a time slot frame with of a data and header frame;

Fig. 6 shows a construction of data and parity blocks from radio link protocol (RLP) frames;

Fig. 7 illustrates the format for a data segment header (DSH) with the number of bits required for each field indicated,

Fig. 8 shows logical layer-1 frame formats assuming 4-level modulation;

Fig. 9 shows logical layer-1 frame formats assuming 8-level modulation with 3 blocks transmitted in each IS-136 time slot;

Fig. 10 shows logical layer-1 frame formats assuming 16-level modulation with 4 blocks transmitted in each IS-136 time slot;

Fig. 11 shows a part of a composite logical flow chart, shown also in Fig. 12, of the process performed at the incremental redundancy soft decoder and FCS checker of Fig 4;

Fig. 12 shows another part of the composite logical flow chart, shown also in Fig. 11, of the process performed at the receive RLP of Fig. 4;

Fig. 13 shows a logical flow chart of the process of generating the feedback signal from the receive RLP and incremental redundancy soft decoder and FCS checker to the Mode 0 RLP of Fig. 4;

Fig. 14 shows a logical flow chart of the process of updating the bit maps at the Mode 0 RLP from the feedback described in Fig. 13;

Fig. 15 shows a logical flow chart of the process of determining the transmission of data and/or parity blocks for each RLP frame without using block bitmaps;

Fig. 16 shows a logical flow chart of the process of determining the transmission of data and/or parity blocks for each RLP frame using block bitmaps and frame bitmaps;

Fig. 17 shows a Mode 0 downlink time slot format;

Fig. 18 shows a Mode 0 uplink time slot format;

Fig. 19 shows uplink ARQ Status block logical format with 2 bits used for channel quality feedback;

Fig. 20 shows mapping of RLP frames to layer-1 data symbols for 4-level, 8-level and 16-level modulation formats on the downlink;

BNSDOCID: <EP___0938207A2_I_>

Fig. 21 shows mapping of RLP frames to layer-1 data symbols for 4-level, 8-level and 16-level modulation formats on the uplink;

Fig. 22 shows a Mode 1 downlink time slot structure; and

Fig. 23 shows a Mode 1 uplink time slot format.

Detailed Description

[0014] Adaptive data rate schemes based on a combination of incremental redundancy coding and adaptive modulation are described earlier for circuit data. Such a channel quality measured technique is described in the commonly owned U.S. patent application SN 08/921454 titled "SYSTEM AND METHOD FOR MEASURING CHANNEL QUALITY INFORMATION; by Krishna Balachandran, Sanjiv Nanda, Srinivas R. Kadaba and Richard P. Ejzak, filed on August 24, 1997, and U. S. patent application SN 08/938031 titled "ADAPTIVE MODIFICATION OF MODULATED AND CODED SCHEMES IN A COMMUNICATION SYSTEM", by Richard P. Ejzak, filed on September 21, 1997, the disclosures of which are hereby incorporated by reference. These schemes achieve high throughput under delay constraints. Provided herein in accordance with the invention is a detailed description of the application of these techniques to 136+ packet data services.

[0015] Described in the application SN 08/938031 by Ejzak is the use of fixed size blocks of size such that an integer number can be accommodated in a slot of a time slotted transmission channel, as a method that enables adaptive modulation. In particular, two, three and four, blocks are accommodated respectively, in a QPSK, 8 PSK and 16 PSK TDMA slot. The requirements of a large per block sequence number to identify the data and parity packets, reduces the efficiency of this method of providing an incremental redundancy, adaptive modulation data communication system using a time slotted communication channel.

[0016] In describing the invention this application uses the media access control (MAC) layer assumptions which are based on the Open System Interconnections (OSI) model. OSI is an internationally accepted frame work of standards for communication between different systems made by different vendors. Most of the dominant communication protocols used today have a structure based on the OSI model. The OSI model organizes the communication process into seven different categories and places these categories in layered sequence based on their relation to the user. Layer 7 through 4 deal with the end to end communication message source and the message destination. While layers 3 through 1 deal with network access.

[0017] Layer 1, the physical layer, deals with the physical means of sending data over lines i.e. the electrical, mechanical and functional control of data circuits. Layer 2, the data link layer, deals with procedures and protocols for operating communication lines. Layer 3, the network layer, determines how data is transferred between computers and routing within and between individual networks.

[0018] It is appreciated that the packet data channel is capable of supporting multiple modulations. The MAC layer is provided with layer 3 frames and translates them into a byte stream using flag delimiters. A radio link protocol (RLP), also referred to as a retransmission link protocol, is used to transfer layer 2 frames between a cell and the mobile station and vice versa. The layer 3 byte stream is segmented into RLP frames, and a sliding window retransmission scheme is used for in-sequence delivery and recovery.

[0019] MAC layer transaction preferably starts with the transmission of a BEGIN frame. On the uplink and downlink, the MAC layer converts the layer 3 frames into a byte stream and packs the byte stream into a series of CONTINUE frames. The last new data burst of a transaction is transmitted using an END frame.

[0020] The BEGIN frame of each transaction is transmitted using 4-level modulation in a stop and wait mode to obtain an acknowledgment from the receiver. On reception of the BEGIN frame, the receiver initializes an RLP. The BEGIN frame is also used to initialize a partial echo (PE) for the transaction, and to specify the mode of operation for subsequent automatic retransmission request (ARQ) mode CONTINUE frames in that transaction.

[0021] There are two possible modes of operation for ARQ mode CONTINUE frames on the downlink and uplink. The first is incremental redundancy (Mode 0) and the second is fixed coding (Mode 1).

[0022] ARQ checks for errors in transmitted data. The sender encodes an error-detection (check) field in the transmitted data based on the contents of the message. The receiver then recalculates the check field and compares it with the check field received. If the check fields match, an ACK (acknowledgment) is transmitted to the sender. If both check fields do not match, a NAK (negative acknowledgment) is returned, and the sender retransmits the message.

[0023] For both uplink and downlink transmissions, bitmap feedback in the form of an ARQ status is provided. In addition, ACK/NAK feedback is provided on a per time slot basis for uplink transmissions.

[0024] For the best tradeoff between high throughput under delay objectives, incremental redundancy or fixed coding is combined with adaptive modulation. Results in the patent applications incorporated by reference herein previously show that if a delay objective is set, 8 or 16-level modulations fail to meet that objective across the $\frac{C}{I+N}$ range of

BNSDOCID: <EP___0938207A2_I_>

interest. As the ratio of signal to interference plus noise decreases a smaller constellation (i.e. level 4 modulation) density is applied. Adaptation to 4 level modulation is employed under delay spread conditions.

[0025] Fig. 1 shows a high level block diagram of operation on the packet data channel 100 in accordance with the invention. A incremental redundancy transmission communication system 102 is shown were layer 3 frames 105 are provided to the layer 2, MAC layer 110, at the transmitter 115 and are translated into a byte stream using flags for demarcation. This permits the MAC layer 110 to provide a unified transport mechanism for different layer 3 protocols. This byte stream is segmented into RLP frames and assigned a frame sequence number (FSN). The FSN is not explicitly transmitted as part of the RLP frame.

[0026] For higher throughput in either mode, Layer 1 120 data is mapped into symbols chosen from a 4-level, 8-level or 16-level modulation based on knowledge of layer 2 backlog and channel quality feedback 125 from the receiver 130. The channel quality is measured in terms of the signal to interference plus noise ratio, $\frac{C}{I+N}$ at the input to the decoder in the layer 2 block 135 via physical layer 140 at the receiver 130. The decoder 135 then outputs the layer 3 frames 145.

[0027] Fig. 2 shows the layer 2 110 functions (i.e, the procedure for mapping layer 3 frames 105 into layer 1 frames 120) of Fig. 1 for Mode 0. In Mode 0, the layer 3 frames 105 are translated by a flag delimiter 150 in a byte stream 155 time slot signal. The byte stream 155 is segmented into fixed length RLP frames and a frame check sequence (FCS) is computed by a computation unit, on the data portion of the RLP frame (excluding the FSN) by an RLP framer 160 (a time slot signal generator). The resulting RLP frame is encoded using, for illustration purposes, a Rate 1/2 systematic convolutional encoder 165. A non-systematic convolutional encoder may alternatively be employed. The sub-slots of data bits 170 and the parity bits 175 from the systematic encoder 165 (also known as both a data block signal generator and parity block signal generator) are interleaved separately in Mode 0 interleavers (a first segmenter) 180 and (a second segmenter) 185, and are blocked, by a blocker 190 (also known as an assembler), into *D* data (a data block signal) and *D* parity blocks (a parity block signal) 195 which are assigned block identifications 1, . . . *D, D+1,* . . . *2D,* respectively. The block sequence number (BSN), also known as a data block signal sequence number, is determined as a combination of the frame sequence number (FSN) and block identification. Fifteen parity bits are punctured in each parity block so that a parity block header containing the block sequence number may be added. Preferably the data blocks do not contain a block header. Depending on whether 4-level, 8-level or 16-level modulation is used, two, three or four blocks (data or parity), outputted from the Mode 0 RLP 200, are respectively combined into a single layer 1 data segment by a MAC and layer 1 field adder (header adder and puncturing unit) 210, which is transmitted in a single IS-136 time slot signal. A data segment header indicating the block sequence number of the first data block in the segment is used to identify data block sequence numbers. The remaining data blocks have consecutive block sequence numbers. Along with the coded data segment header (CDSH), other MAC and physical layer headers such as modulation type and partial echo (PE), are added in each time slot signal.

[0028] In Mode 1, Fig. 3, the layer 3 frames 105 are translated by a flag delimiter (flag marker 150) in a byte stream 155. The byte stream 155 is segmented into fixed length RLP frames and a FCS is computed on the data portion of the RLP frame by Mode 1 RLP framer 161. The resulting RLP frames are processed by the Mode 1 RLP 220. Depending on whether 4-level, 8-level or 16-level modulation is used, two, three or four consecutive RLP frames, outputted from the Mode 1 RLP 220, are respectively combined into a single data segment by a data segment and FCS combiner 230 (also called header and FCS adder). The FCS computed on this data segment and a data segment header indicating the FSN of the first RLP frame are added in header and FCS adder 230.

[0029] Then, the header, data and FCS are encoded using a Rate ½ convolutional encoder 235. The output of the encoder 235 is run through a Mode 1 interleaver 240 and additional MAC and physical layer header fields are added by a header adder 210. The output constitutes layer 1 data 120 which is transmitted in a single time slot. The data segment header is transmitted as part of the encoded layer 1 data segment in this case.

[0030] The advantage of incremental redundancy (Mode 0) is that it achieves a higher throughput because redundant bits are transmitted when necessary. The combination of incremental redundancy and adaptive modulation is proposed for both downlink and uplink directions.

[0031] Fig. 4 shows both the transmitting (shown in Fig. 1 as layer 2 functions) and receiving functions for the Mode 0 procedure of Fig. 2. The procedure described at transmitter 115 from the receipt of layer 3 frames 105 to the output 195 of blocker 190 is the same as described with reference to Fig. 2. The layer 1 frames 120 are output from the header adder 210 and input into an adaptive modulator 250. The adaptive modulator 250 modulates and transmits the layer 1 frames 120 through the packet data channel 100 to the channel demodulator 255 at the receiver 130. The channel demodulator 255 demodulates the modulated layer 1 frames 120 received from the packet data channel 100 and input the results to a receiver RLP 260. The receiver RLP 260 decodes the data segment header to determine the sequence numbers of the received data blocks, as well as the number of parity blocks. The receiver RLP 260 also decodes the parity block header to determine the sequence numbers of the parity blocks. If any of the headers of data on parity blocks are decoded correctly, the data is provided to the incremental redundancy soft decoder and FCS checker (IRSD) 265; and ACK/NAK (acknowledgment/non-acknowledgment) feedback 270 is provided to the transmit RLP 275 indi-

BNSDOCID: <EP___0938207A2_I_>

cating acknowledgment of the data and parity blocks. The transmit RLP 275 and modulation decision unit 278 is a functional block that represents the functions of Mode 0 RLP 200 of Fig. 2.

[0032] The IRSD 265 receives the output from the receive RLP 260 and produces three outputs. The first output is a decoded data sequence of the received layer 1 frames 120 that is input into a flag remover 280. The second is an ACK/NAK feedback 270 output to the transmit RLP 275. The third output is a channel quality feedback signal 125, which is transmitted back to the transmit RLP 275 and modulation decision unit 278. The flag remover 280 removes the flags from the received data and outputs the received layer 3 frames 285.

[0033] Fig. 5 shows the format of the RLP frame 290, output or the Mode 0 RLP framer 160, which is constructed by adding a FCS 295 to RLP data, R, 300. The FCS 295 is a cyclic redundancy check (CRC) of length L_{DCRC} 305 computed over the data bits. The FCS 295 is added to each RLP frame 300 which consists of L_{DATA} 310 data bits. The lengths L_{DATA} 310 and L_{DCRC} 305 are design parameters. The frames of length L_{FRAME} 315 = L_{DATA} 310 + L_{DCRC} 305 are passed to the encoder 165 of Fig. 2.

[0034] Fig. 6 shows that each RLP frame 290, of size L_{FRAME} 315, Fig. 5, bits, is encoded using a binary rate 1/2 systematic convolutional code by encoder 165. The rate 1/2 binary convolutional encoder 165 is preferably a 32 state maximum free distance code with octal generators. There are L_{FRAME} 315, Fig. 5, data bits 170, and L_{FRAME} 315 parity bits 175, at the output of the encoder 165, Fig. 6, which are divided into blocks as described as follows.

[0035] The L_{FRAME} 315, Fig. 5, data bits 170 at the output of the encoder 165 are segmented into D blocks of length $L_{BLOCK} = \frac{L_{FRAME}(315)}{D}$. These blocks are called data blocks 320 which are denoted by D_{ij} 320 ($j = 1 \ldots D$). The parameter D determines the delay and/or throughput performance as described in the patent applications incorporated by reference herein. The parity bits 175 are punctured and segmented into D parity blocks 325 of equal size denoted by P_{ij} 325 ($j = 1 \ldots D$ as an example P_{i1} 330, P_{i2} 335 and P_{i3} 340). It is appreciated that puncturing of outputs of a convolutional encoder is a procedure of deleting a set of predetermined bits to decrease the number of parity bits. Of the L_{FRAME} 315 parity bits 175 at the encoder 165 output, Dh parity bits are punctured (h in each of the D parity blocks 325); h is the size of the header required for each parity block.

[0036] To each parity block 330, 335 and 340, a parity/control block header (PCBH) 345, 350 and 355 is appended to the front of the parity block P_{ij} 325. For parity blocks, PCBH 345, 350 and 355 contains a one bit PCBH header type (=1 for parity block) and a ten bit block sequence number (BSN). The PCBH 345, 350 and 355 is selectively encoded using a Hamming code.

[0037] The data blocks, Di_1 to D_{iD} 320 contain no redundancy and represent a one to one mapping to the frame R_i 290. The parity blocks Pi_1 to P_{iD} 325 contain parity information derived from R_i 290 and are used by the protocol for forward error correction (FEC) upon decoding failure at the receiver 130 of Fig. 1. Fig. 6 is illustrative of the mapping of data 170 and parity 175 bits to blocks 320 and 325 when D=3 (i.e., with three data blocks per frame). A separate header for each data block 320 is not required. The data 320 and parity 325 blocks are derived from R_i 290 by using a systematic rate 1/2 binary convolutional encoder 165 as shown. For a non-systematic convolutional encoder the same procedure is used. The output bits of the encoder 165 are mapped to the blocks 320 and 325 in an interleaved manner to maximize time and code diversity.

[0038] The Mode 0 RLP 200 of Fig. 2 delivers data in-sequence by initially sending just the RLP data blocks $D_{i1} \ldots D_{iD}$ 320 followed by sending additional RLP parity blocks P_{ij} 325 whenever the receiver 130 of Fig. 1 fails to decode the RLP frame R_i 290 of Fig. 5 correctly. The transmitter 115 of Fig. 1 cannot discard the data 170 and parity 175 bits corresponding to a RLP frame R_i 290 until it has received a positive acknowledgment from the receiver 130, Fig. 1, for R_i 290. The protocol operates by maintaining in a table, frame table and block table, a bitmap for blocks that have been transmitted but not yet acknowledged. Upon obtaining feedback from the receiver 130, Fig. 1, the transmitting protocol updates the table and decides on which data 320 and/or parity 325 blocks to transmit in subsequent time slots.

[0039] Control blocks may be inserted instead of parity blocks 325 in the time slot. For each control block, the PCBH 345 header type bit is set to 0. Examples of control blocks include ARQ Status Block - indicates a bit map of received blocks along with the last in-sequence received block sequence number and END Block - used to indicate completion of a transmission.

[0040] The receive RLP 260 and the IRSD 265 of Fig. 4 is the peer of the Mode 0 RLP 200 of Fig. 2 and is responsible for combining the received blocks D_{ij} 320 and P_{ij} 325, Fig. 6, and jointly decoding them to recover the layer 2 frame R_i 290. The receive RLP 260, Fig. 4, maintains a table to store the received data and parity blocks and decoded RLP frames 300, Fig. 6, until they are delivered in-sequence to layer-3 as layer 3 frames 285, Fig. 4. The blocks are retrieved from layer-1 in soft-decision format and to reduce memory overhead are stored in the receive table by quantizing to q levels.

[0041] The incremental redundancy decoder 265, Fig. 4, corresponds to the transmit encoder 165 of Fig. 4. It is passes the received subset of the soft-decision blocks D_{ij} and P_{ij} and attempts to decode using preferably a soft decision Viterbi algorithm. Any missing soft bits are treated as erasures by the decoder 265 of Fig. 4. The output of the Viterbi decoder is passed to the FCS decoder which computes the FCS and indicates to the receive RLP 260 whether the received frame R_i 290 passed the frame check.

BNSDOCID: <EP___0938207A2_I_>

[0042] The data bit stream 170, Fig. 4, and parity bit stream 175 at the encoder output 165 are separately interleaved, by Mode 0 interleavers 180 and 185, and segmented into data and parity blocks by blocker 190. The two, three or four (combinations of data and parity/control) blocks, outputted by the Mode 0 RLP 200, are transmitted in each time slot corresponding to the use of 4-level, 8-level or 16-level modulation respectively. Data or parity blocks are selectively mapped directly into the Layer 1 data portion of an IS-136 time slot (DATA fields shown in Fig. 11 and 12). Figs. 8, 9 and 10 illustrate the logical Layer 1 frame 120, Fig. 5, formats as a function of the modulation type.

[0043] Fig. 7 illustrates that a Layer 1 frame 120, Fig. 4, is associated with a data segment header (DSH) 360. The DSH 360 contains a block sequence number (BSN) 365 which preferably consists often bits. The BSN 365 is associated with the first data block in a time slot; if there are no data blocks, then the BSN 365 is given a default value which is ignored by the receiver 130 of Fig. 4. The DSH 360 also includes a poll indicator (PI) 370, Fig. 7, which preferably consists of one bit. The PI 370 is used by the system 102, Fig. 1, to request an ARQ status frame. The DSH 360 also includes parity/control block pointer (PCBP) 375 which preferably consists of two bits. The PCBP 375 indicates the composition of the time slot in terms of data and parity/control blocks. The DSH 360 also has a reserved (RSVD) 380 header which is preferably one bit in length for future needs.

[0044] The 14 bit DSH 360, Fig. 7, is encoded to obtain a 22 bit coded DSH (CDSH) 385 of Fig. 8, 9 and 10. The encoding preferably employs Hamming codes. In addition each time slot contains a 12 bit coded partial echo (PE) assigned to the transaction in progress which identifies the recipient of the data on the downlink or the originator on the uplink.

[0045] On the downlink, each RLP frame of size 336 bits (40 octet RLP data frame + 16 bits CRC) is encoded using a rate 1/2, memory five, systematic convolutional code. The number of data blocks per RLP frame *(D)* is set to a nominal value of three. The output of the encoder is segmented into three data blocks, each of size 112 bits, and three parity blocks, each of size, 97 bit (punctured from 112 bits). A 15 bit header parity/control block (PCBH) 390, Figs. 8, 9 and 10, is used for each parity block. These blocks are carried two, three or four per time slot as discussed above.

[0046] On the uplink, each frame of size 342 bits (41 octet RLP frame + 14 bits CRC) may be encoded using a rate 1/2, memory five, systematic convolutional code. The number of data blocks per RLP frame *(D)* is set to three. The output of the encoder is segmented into three data blocks, each of size 114 bits, and three parity blocks, each of size, 99 bits (punctured from 114 bits). A 15 bit header (PCBH) 390, Figs. 8, 9 and 10, is used for each parity block. These blocks are carried two, three or four per time slot as discussed above.

[0047] Fig. 8 shows the logical layer-1 frame formats assuming level 4 modulation with two blocks transmitted in each IS-136 time slot. The PCBP 375 value in the DSH 360, Fig. 7, is used to identify the composition of the data segment in the time slot in terms of data and parity blocks. As an example, PCBP 375 = 00 indicated two parity blocks, PCBP 375 = 01 indicates one data and one parity block and PCBP 375 = 10 indicates two parity blocks. Fig. 9 shows the use of PCBP and the composition of time slot for 8 level modulation and Fig. 10 is for 16 level modulation.

[0048] Fields such as adaptation field (AF) 395, Figs. 8, 9 and 10, and DSH 360, Fig. 7, assume special positions in the time slot and are not transmitted as part of the layer-1 data. The block sequence number (BSN) 365, Fig. 6, in the CDSH 385, Figs. 8, 9 and 10, denotes the sequence number corresponding to the first data block 320 in the slot. If only parity blocks 340 are transmitted, this field is ignored. In addition to the fields shown above, the coded PE assigned to the MAC Layer transaction is used as a mobile station identifier.

[0049] Figs. 11, 12, 13 and 14 are illustrative of examples of the process performed by the receive RLP 260, Fig. 4, IRSD 265, feedback 270 to the Mode 0 RLP 275, and the Mode 0 RLP 275. In step 400 of Fig. 11, the receive RLP 260, Fig. 4, receives a demodulated time slot from the channel demodulation 255 and in step 405, Fig. 11, decodes the data segment header (DSH) 360 of Fig. 7. In decision step 410, if the DSH 360, Fig. 7, is successfully decoded the process continues to step 415, Fig. 11. If the DSH 360, Fig. 7, is not successfully decoded the process instead continues to step 420, Fig. 11. In step 420, the receive RLP 260, Fig. 4, discards the time slot and the process continues to node A 425, Fig. 11, which starts the new process at the IRSD 265 of Fig. 4.

[0050] If the DSH 360, Fig. 7, is successfully decoded the receive RLP 260, Fig. 4, in step 415, Fig. 11, reads the PCBP 375, Figs. 7, 8, 9 and 10, in the DSH 360, Fig. 7, and determines the number of data 320, Fig. 6, and parity blocks 325 and decodes the PCBH 345, 350, 355 Fig. 6, 390 Figs. 8, 9 and 10 for the parity blocks. The receive RLP 260, Fig. 4, then, in step 430, Fig. 11, updates the block bitmap table for the received data and parity blocks and in step 435 passes the soft information and data/parity block sequence numbers to the IRSD 265 and continues the process at the IRSD 265 at node A 425.

[0051] Fig. 12 shows the process performed by the IRSD 265 of Fig. 4. The process starts in step 440, Fig. 12, the IRSD 265, Fig. 4, receives soft information and data/parity block numbers from the receive RLP 260. In step 445, Fig. 12, the IRSD 265, Fig. 4, decodes the RLP frames associated with the new received data and/or parity blocks using previously stored data and/or parity blocks associated with the RLP frames. The IRSD 265, in step 450, Fig. 12, then checks the FCS associated with each RLP frame decoded. In decision step 455, if the FCS check is successful the process continues to step 460. If the FCS check is not successful the process instead continues to step 465. In step 465, the new soft information associated with the received data/parity block is stored and the process ends at 470.

BNSDOCID: <EP___0938207A2_I_>

[0052] If the FCS check is successful, the IRSD 265, Fig. 4, in step 460. Fig. 12, updates the table from the RLP frames and in step 475 stores the successfully decoded frames in sequence buffer. In step 480, the IRSD 265, Fig. 4, passes the in-sequence data to the flag remover 280 and updates the frame bit map table and ends the process at 470, Fig. 12.

[0053] Fig. 13 illustrates the process of generating the feedback signal 270 of Fig. 4. First the receive RLP 260, Fig. 4, obtains the frame bit map from the frame bit map table in step 485, Fig. 13. The receive RLP 260, Fig. 4, then obtains the data and/or parity block bit map from the block bit map table in step 490, Fig. 13, and transmits the bit maps via feedback signal 270, Fig. 4, to the mode 0 RLP 275 in step 495, Fig. 13.

[0054] Fig. 14 is illustrative of the mode 0 RLP 275, Fig. 4, processing the feedback signal 270. In step 505, the mode 0 RLP 275, Fig. 4, receives the bit maps transmitted by the receive RLP 260. The mode 0 RLP 275 then updates the RLP frame bit maps in step 510. Fig. 14, and updates the data and/or parity block bit maps in step 515.

[0055] Fig. 15 illustrates the process of transmitting data and/or parity blocks for each RLP frame if only frame bit map are used. In step 525, the mode 0 RLP 275, Fig. 4 receives a new encoded blocked RLP frame and in step 530 orders the block of the RLP frame into a transmit sequence. As an example a round robin sequence of D_{i1}, D_{i2}, D_{i3}, P_{i1}, P_{i2} and P_{i3} is used where D_{i1} repeats after P_{i3}. In step 535, the mode 0 RLP 275, Fig. 4, transmits the data blocks D_{i1}, D_{i2} and D_{i3}. In decision step 540, if the frame bit map indicates a frame acknowledgement (ACK) at the receive RLP 260 Fig. 4 the process continues to step 545. If instead a NAK is registered at the receive RLP 260, Fig. 4, the process continues to step 550. In step 550, the mode 0 RLP 275, Fig. 4, discards the transmit sequence and the process ends at 555.

[0056] At step 545 the mode 0 RLP 275, Fig. 4, transmits the next data or parity block in the transmit sequence and waits in step 560 for the receive RLP 260 to respond. The process then continues back to decision step 540 and the process continues unit all the blocks are properly transmitted.

[0057] Fig. 16 illustrates the process of transmitting data and/or parity blocks for each RLP frame if both a block and a frame bit map are used. In step 565, the mode 0 RLP 275, Fig. 4, receives a new encoded blocked RLP frame and in step 570 orders the block of the RLP frame into a transmit sequence. As an example a round robin sequence of D_{i1}, D_{i2}, D_{i3}, P_{i1}, P_{i2} and P_{i3} is used where D_{i1} repeats after P_{i3}. In step 575, the mode 0 RLP 275, Fig. 4, transmits the data blocks D_{i1}, D_{i2} and D_{i3}. In decision step 580, if the frame bit map indicates a frame acknowledgement (ACK) at the receive RLP 260, Fig. 4, the process continues to step 585. If instead a NAK is registered at the receive RLP 260, Fig. 4 the process continues to step 590. In step 590, the mode 0 RLP 275, Fig. 4, discards the transmit sequence and the process ends at 595.

[0058] At decision step 585, if the block bit map indicates a block non-acknowledgement (NAK) at the receive RLP 260. Fig. 4 the process continues to step 600. If instead, decision step 585 result in ACK, the process continues to step 605. At step 605, the mode 0 RLP 275, Fig. 4, transmits the next data or parity block in the transmit sequence and waits in step 610 for the receive RLP 260 to respond. The process then continues back to decision step 580 and the process continues unit all the blocks are properly transmitted.

[0059] If a NAK is the result of decision step 585, the process proceeds to step 600. At step 600, the mode 0 RLP 275, Fig. 4, transmits NAK data or parity block without a transmit sequence and waits in step 610 for the receive RLP 260 to respond. The process then continues back to decision step 580 and the process continues unit all the blocks are properly transmitted.

[0060] Fig. 17 shows a downlink time slot format for Mode 0. The time slot has the following fields: SYNC - synchronization field (SYNC) 615, packet channel feedback (PCF) 620, coded superframe phase (CSFP) and coded adaptation field (CAF) (shown as CSFP/CAF 625), coded partial echo (CPE) 630, coded data segment header (CDSH) 635, 385, Figs. 8, 9 and 10, reserved (RSVD) 640 and data block 320 Figs. 6, 8, 9 and 10.

[0061] The time slot structures for the downlink and uplink preferably use differentially encoded PSK (DPSK) constellations. If coherent PSK is used, pilot symbols and reference symbols are inserted at appropriate positions in the time slot structure.

[0062] The PCF 620 fields are used as a mechanism for providing ACK/NAK 270, Fig. 4, and channel quality feedback 125, Fig. 4, and time slot assignments on the reverse packet data channel. The CAF 625 on the downlink consists of two bits which indicate the modulation format (4-level, 8-level or 16-level) used for the data. The layer-1 data fields consist of two, three or four blocks which are constructed using the procedure described earlier in Figs. 8, 9, 10 and 11. The adaptation field, superframe phase (SFP) and one reserved bit are encoded together using the same Hamming code that is used for CDVCC in IS-136, 22 bits are used for the CDSH field 635, and a 12 bit CPE 630 field is used to identify the recipient of the data. The SYNC 615 field is transmitted using $\pi/4$-DQPSK. The CSFP/CAF 625, PCF 620, CPE 630 and CDSH 635 are also transmitted using 4-level modulation.

[0063] Contention accesses are transmitted using 4-level modulation. Higher level modulations (8-level and 16-level) are used by mobile stations which have more than one burst to transmit in ARQ CONTINUE mode, and are sniffing for reservation based transmission opportunities.

[0064] Figure 18 shows the preferred uplink time slot structure. The guard (G) 645, ramp (R) 650, preamble (PREAM)

BNSDOCID: <EP___0938207A2_I_>

655 and SYNC 615 fields are maintained as in IS-136; a new coded adaptation 20 field (CAF) 660 is introduced to indicate the modulation format for the data. Two bits are used for the modulation type; these are encoded using a (6,2) code. The CDSH 635 uses 22 bits, and a 12 bit CPE 665 is used to identify the mobile station. G 645, R 650, PREAM 655, CPE 665, CDSH 635 and CAF 660 are transmitted using 4-level modulation.

[0065] The modulation type is fixed for a time slot and not over an RLP frame or an entire MAC Layer transaction. Layer 1 determines the modulation type to use for the next time slot based on the channel quality feedback 125, Fig. 4, from the receiver 130 and the needs of layer 2. The modulation formats utilized are 4 level (DQPSK or $\pi/4$-coherent PSK), 8 level (differential or coherent) and 16 level (PSK, DPSK or QAM) modulation. A two bit adaptation field is used to indicate the coding and/or modulation format used. Table 1 shows the mapping of the AF values to corresponding modulation formats.

[0066] A default value specifies the default DQPSK format with Rate 5/6 coding within the time slot to allow different packet data channel frame formats and interworking between incremental redundancy and coding modes.

Table 1

AF	Coding and/or Modulation Format
00	4-level ($\pi/4$-QPSK or $\pi/4$-DQPSK)
01	8-level (PSK or DPSK)
10	16-level (QAM, PSK or DPSK)
11	Default (e.g. rate 5/6, memory 5 convolutional coding with 4-level modulation)

[0067] Channel Quality Feedback (CQF) 125, Fig. 4, is provided by the mobile station on the reverse packet data channel in a layer-2 "ARQ Status" block as shown in Fig. 19. The ARQ status block is preferably transmitted in any reserved uplink time slot. A two bit channel quality indicator is used to denote the maximum allowable modulation type.

[0068] The PCBH 670 for the ARQ status block consists of one bit PCBH header type (= 0), one bit control block type (= 0, for ARQ status block), two bit CQF field, seven bit frame sequence number (FSN), identifying the beginning frame for the bit-map feedback.

[0069] The PCBH 670 is encoded using the (15,11) code. The rest of the ARQ status block consists of the bit-map feedback 675 and CRC 680, encoded using the default 5/6 rate convolutional code.

[0070] For the uplink, channel quality feedback is provided in the PCF corresponding to reserved slots. The subchannel feedback (SF) field in the PCF provides ACK/NAK and channel quality feedback for the previous slot as well as channel quality feedback (CQF) which serves as a modulation format advisory for subsequent assignments to the mobile station being acknowledged. For ACK/NAK feedback, a simple (6,1) repetition code is used A (6,2) code is used for channel quality feedback.

[0071] The receiver 130, Fig 4, obtains channel quality information by averaging the Euclidean distance between the received symbol sequence and the decoded sequence of symbols to obtain an implicit measure of the signal to interference plus noise ratio, C/(I+N) at the input to the decoder 265. As described in the patent applications incorporated by reference herein, this method yields good estimates under both noise limited and interference limited conditions for different coded modulation schemes across the range of Doppler frequencies. The metric computation is independent of the actual decoder implementation and can be computed using the decoded information sequence.

Table 2

Channel Conditions	Channel Quality Indicator - Maximum Allowable Modulation Format
$C/(I+N) < \theta_L$	00 (4-level)
$\theta_L < C/(I+N) < \theta_H$	01 (8-level)
$C/(I+N) > \theta_H$	10 (16-level)

[0072] Table 2 shows the scheme for determining the maximum allowable modulation type based on C/(I+N) estimated at the receiver. The mobile station receiver uses the channel quality metric along with knowledge of C/(I+N) thresholds, θ_L, θ_H and which are transmitted over the packet broadcast control channel (PBCCH).

[0073] As in Mode 0, the byte stream obtained from Layer 3 frames is segmented into RLP frames. However, in this mode, RLP frames are smaller in size than in the incremental redundancy case. Two, three or four RLP frames are transmitted in each time slot depending on the modulation type (i.e., if 4-level, 8-level or 16-level modulation respectively, is used).

[0074] Two, three or four RLP frames are concatenated with a ten bit frame check sequence (FCS) and a data

BNSDOCID: <EP___0938207A2_I_>

segment header (DSH) which contains a ten bit block sequence number (BSN) and one bit poll indicator (PI) to request ARQ status

[0075] The resulting data segment, containing RLP frames, DSH, and CRC bits, is encoded using a memory 5, rate 1/2 convolutional code. The coded bits are punctured and mapped to symbols that are chosen from a 4-level, 8-level or 16-level constellation. The puncturing depends on the modulation format used for the transmission of that slot as shown in Figs. 20 and 21.

[0076] Since the frame check is on the data segment consisting of two, three, or four RLP frames, if the CRC fails at the receiver, then all frames in the slot will be lost and will have to be recovered through ARQ recovery. Assigning a FCS to each "small" RLP frame is inefficient due to excessive overhead.

[0077] On the downlink, each RLP frame consists of 88 bits (eleven octets). To each set of two, three or four RLP frames, an eleven bit DSH and ten bit CRC are added. The resulting combination is encoded using a rate 1/2, memory 5 convolutional code as shown in Figure 20. At the encoder output, 128, 201 or 254 bits are punctured depending on whether 4-level, 8-level or 16-level modulation is used, and the remaining bits are mapped to the desired modulation format, and inserted in the DATA fields in the time slot.

[0078] On the uplink each RLP frame consists of 96 bits (12 octets). To each set of two, three or four RLP frames, an eleven bit DSH and ten bit CRC are added. The resulting combination is encoded using a rate 1/2, memory 5 convolutional code as shown in Fig. 21. At the encoder output, 176, 243 or 310 bits are punctured depending on whether 4-level, 8-level or 16-level modulation is used, and the remaining bits are mapped to the desired modulation format.

[0079] The slot formats assume the use of DPSK. If coherent PSK or QAM are used, then pilot symbols are inserted at appropriate positions in the time slot.

[0080] The downlink time slot format is shown in Figure 22. The packet channel feedback (PCF) 800 fields are used as a mechanism for providing ACK/NAK and channel quality feedback and time slot assignments on the reverse packet data channel. The layer-1 data fields consist of two, three or four RLP frames which are constructed using the procedure described above. The adaptation field (AF) on the downlink consists of two bits which indicate the modulation format (4 level, 8 level or 16 level) used for the data. The adaptation field, superframe phase (SFP) and one reserved bit are encoded together using the same (12,8) Hamming code that is used for CDVCC in IS-136. A 12 bit CPE 810 field is used to identify the recipient of the data 805. The SYNC 805 field is transmitted using $\pi/4$-DQPSK. The CSFP/CAF 820, PCF 800 and CPE 810 are transmitted using 4-level modulation.

[0081] Fig. 23 shows the proposed uplink time slot structure. The guard (G) 825, ramp (R) 830, preamble (PREAM) 835 and SYNC 815 fields are maintained as in IS-136; a new coded adaptation field (CAF) 840 is introduced to indicate the modulation format for the data 845. A 12 bit CPE 810 is used to indicate the transmitter identity. Two bits are used for the modulation type; these are encoded using a (6,2) code. G 825, R 830, PREAM 835, CPE 810 and CAF 840 are transmitted using 4-level modulation.

[0082] Contention accesses are transmitted using 4-level modulation. Higher level modulations (8-level and 16-level) are used in ARQ Mode CONTINUE frames by mobile stations which have more than one burst to transmit, and are sniffing for reservation based transmission opportunities.

[0083] Adaptive modulation as in Mode 0, the modulation type is fixed only for a time slot and not over an entire MAC Layer transaction. Layer 1 determines the modulation type to use for the next time slot based on the channel quality feedback from the receiver and the needs of Layer 2. Since the RLP frame sizes in Mode 1 are chosen to be the same independent of the modulation type, retransmitted frames may be sent on a different modulation type from the original transmission. The modulation formats considered are 4 level (DQPSK or $\pi/4$-coherent PSK), 8 level (differential or coherent) and 16 level (PSK, DPSK or QAM) modulation. A two bit adaptation field is used to describe the coding and/or modulation format used. Table 1 shows the mapping of the AF values to corresponding modulation formats.

[0084] A default value specifies the default DQPSK format with Rate 5/6 coding. This permits different packet data channel frame formats and interworking between incremental redundancy and coding modes.

[0085] Similar to Mode 0, the downlink Channel Quality Feedback (CQF) is provided by the mobile station on the reverse packet data channel in a layer-2 "ARQ Status" frame. The ARQ status frame may be transmitted in any reserved uplink time slot. A two bit channel quality indicator is used to denote the maximum allowable modulation type, and the cell transmitter uses this information along with knowledge of the Layer-2 data backlog in order to determine the modulation to use in subsequent slots.

[0086] For the uplink, channel quality feedback is provided in the PCF corresponding to reserved slots. The subchannel feedback (SF) field in the PCF provides ACK/NAK and channel quality feedback for the previous slot as well as channel quality feedback (CQF) which serves as a modulation format advisory for subsequent assignments to the mobile station being acknowledged. For ACK/NAK feedback, as in Mode 0, a simple (6,1) repetition code is used. A (6,2) code is used for channel quality feedback.

[0087] Stated generally, the present invention encompasses an incremental redundancy transmission communication system 102, Fig. 2. The system comprises a time slot signal generator 160 that creates a time slot signal 155

BNSDOCID: <EP___0938207A2_I_>

having at least one sub-slot with a fixed size and a data block signal generator 165 that creates a data block signal 170 sized to fit in the time slot. A header adder 210 that generates a header having a data block signal sequence number in the header for the time slot signal is also provided. A receiver 130, Fig. 4,is also provided that identifies in the header for the time slot signal, a number of data block signals 170 consecutively transmitted.

[0088] The invention further includes a parity block signal generator 165, Fig. 2, that creates a parity block signal 175 sized smaller than the data block signal 170 such that the parity block signal 175 and the data block signal sequence number fit within the sub-slot, and a transmitter 115 that transmits the data block signal 170 and parity block signal 175 in the sub-slot within the time slot signal.

[0089] The present invention further encompasses a method for incremental redundancy transmission in a communication system that uses radio (or retransmission) protocol link frames. The method comprises the step of creating a time slot signal having at least one sub-slot with a fixed size, creating a data block signal 170, Fig. 2, sized to fit in the sub-slot, establishing a header having a data block signal sequence number in the header for the time slot signal and identifying in the header for the time slot signal, the number of data block signals 170 consecutively transmitted. The steps of providing, in a header of the time slot signal, a data block sequence number for a first data block signal 170 in a sequence of data block signals 170 and creating a parity block signal 175 smaller than the data block signal 170 such that the parity block signal 175 sequence number fits within the sub-slot signal are also provided.

[0090] While the specification in this invention is described in relation to certain implementations or embodiments, many details are set forth for the purpose of illustration. Thus, the foregoing merely illustrates the principles of the invention. For example, this invention may have other specific forms without departing from its spirit or essential characteristics. The described arrangements are illustrative and not restrictive. To those skilled in the art, the invention is susceptible to additional implementations or embodiments and certain of the details described in this application can be varied considerably without departing from the basic principles of the invention. It will thus be appreciated that those skilled in the art will be able to devise various arrangements which, although not explicitly described or shown herein, embody the principles of the invention are thus within its spirit and scope.

Claims

1. An incremental redundancy transmission communication system comprising:

   a time slot signal generator that creates a time slot signal having at least one sub-slot with a fixed size;

   a data block signal generator that creates a data block signal sized to fit in the at least one sub-slot;

   a header adder that generates a header having a data block signal sequence number in the header for the time slot signal; and

   a receiver that identifies in the header for the time slot signal a number of data block signals consecutively transmitted.

2. The system of claim 1 including a parity block signal generator that creates a parity block signal sized smaller than the data block signal such that the parity block signal and the data block signal sequence number fit within the sub-slot, and a transmitter that transmits the data block signal and parity block signal in the sub-slot within the time slot signal.

3. The system of claim 2 wherein the receiver identifies in the header for the time slot signal the number of data block signals and parity block signals in the time slot signal.

4. The system of claim 2 wherein the data block signal generator and parity block signal generator create an integer number of total data block signals and the parity block signals that fit in the time slot signal.

5. The system of claim 2 including an encoder that separately encodes the sequence of data block signals and parity block signals, and an interleaver that separately interleaves the sequence of data block signals and parity block signals.

6. The system of claim 5 further including an assembler that assembles the sequence of data block signals and parity block signals into time slot signals as determined by a retransmission protocol after to the encoder and interleaver.

BNSDOCID: <EP___0938207A2_I_>

7. The system of claim 1 including a transmitter that transmits a parity block signal with the data block signal sequence number in the sub-slot.

8. The system of claim 1 including a receiver that identifies the sequence of data block signals in the time slot as being consecutive from a first data block.

9. The system of claim 1 including a transmitter that sizes the data block signals for operation in a plurality of modulation schemes.

10. The system of claim 9 wherein the transmitter selectively sizes the data block signals so that the data block signals fit within the sub-slot at each one of the plurality of modulation schemes.

11. The method of claim 10 wherein the transmitter uses a code rate defined by the equation

$$\frac{ND}{(ND + MP)} = \frac{N}{(2N - q)}$$

in which N is the data block signal size, M is a parity block signal size, q is the length of the data block signal sequence number field, D is the number of data blocks and P is the number of parity blocks.

12. The system of claim 1 including a computation unit that computes a frame check sequence, and a adder that adds the frame check sequence to the radio link protocol frame to define a frame Bi.

13. The system of claim 12 including an encoder that encodes the frame Bi into N data bits and N parity bits, a first segmenter that segments the N data bits into D data block signals, a second segmenter that segments the N parity bits into P parity block signals of equal size, and a puncturing unit that selectively punctures the P parity block signals to insert a header.

14. The system of claim 13 wherein the header includes a header type identifier and a data block signal sequence number for selected parity block signals.

15. The system of claim 13 further includes a memory unit that stores information corresponding to the N data bits and N parity bits.

16. The system of claim 15 wherein the memory unit includes a bitmap for delivered D data block signals and P parity block signals.

17. A method for incremental redundancy transmission in a communication system that uses radio protocol frames, comprising the steps of:

creating a time slot signal having at least one sub-slot with a fixed size;
creating a data block signal sized to fit in the sub-slot;
establishing a header having a data block signal sequence number in the header for the time slot signal; and
identifying in the header for the time slot signal the number of data block signals consecutively transmitted.

18. The method of claim 17 including the steps of creating a parity block signal sized smaller than the data block signal such that the parity block signal and the data block signal sequence number fit within the sub-slot and transmitting the data block signal and parity block signal in the sub-slot within the time slot signal.

19. The method of claim 17 including the step of transmitting the parity block signal with the data block signal sequence number in the sub-slot.

20. The method of claim 17 including the step of identifying the sequence of data block signals in the time slot as consecutive from a first data block.

21. The method of claim 17 including the step of sizing the data block signals for operation in a plurality of modulation schemes.

BNSDOCID: <EP___0938207A2_I_>

22. The method of claim 21 including the step of selectively sizing of the data block signals so that the data block signals fit within the sub-slot at each one of the plurality of modulation schemes.

23. A method for incremental redundancy transmission in a communication system, comprising the steps of:

creating a time slot signal having a sub-slot with a fixed size;
creating a data block signal sized to fit within the sub-slot;
providing, in a header of the time slot signal, a data block signal sequence number for a first data block signal in a sequence of data block signals; and
creating a parity block signal smaller than the data block signal such that the parity block signal and the data block signal sequence number fit within the sub-slot signal.

24. The method of claim 23including the step of transmitting the parity block signal with the data block signal sequence number in the sub-slot.

25. The method of claim 23 including the step of identifying the sequence of data block signals in the time slot signal as consecutive from the first data block signal.

26. The method of claim 18 or 23 including the step of identifying in the header for the time slot signal the number of data block signals and parity block signals in the time slot signal.

27. The method of claim 26 including the step of sizing the data block signals for operation in a plurality of modulation schemes.

28. The method of claim 27 including the step of selectively sizing of the data block signals so that the data block signals fit within the sub-slot at each one of the plurality of modulation schemes.

29. The method of claim 18 or 23 including the step of inserting into the time slot signal an integer number of total data block signals and parity block signals.

30. The method of claim 22 or 29 wherein the transmitter uses a code rate defined by the equation

$$\frac{ND}{(ND + MP)} = \frac{N}{(2N - q)}$$

in which N is the data block signal size, M is a parity block signal size, q is the length of the data block signal sequence number field, D is the number of data blocks and P is the number of parity blocks.

31. The method of claim 18 or 23 involving the steps of separately encoding the sequence of data block signals and parity block signals, and interleaving separately the sequence of data block signals and parity block signals.

32. The method of claim 18 or 31 including the step of assembling the sequence of data block signals and parity block signals into time slot signals as determined by a retransmission protocol prior to the steps of encoding and interleaving.

33. The The method of claim 32 including including the steps of decoding the sequence of data block signals and parity block signals, and decoding the data stored at the transmitter only when a positive acknowledgment is received that the time slot signal is correctly decoded.

34. The method of claim 17 or 33 including the steps of computing a frame check sequence, and adding the frame check sequence to the radio link protocol frame to define a frame Bi.

35. The method of claim 34 including the steps of encoding the frame Bi into N data bits and N parity bits, segmenting the N data bits into D data block signals, segmenting the N parity bits into P parity block signals of equal size, and selectively puncturing the P parity block signals to insert a header.

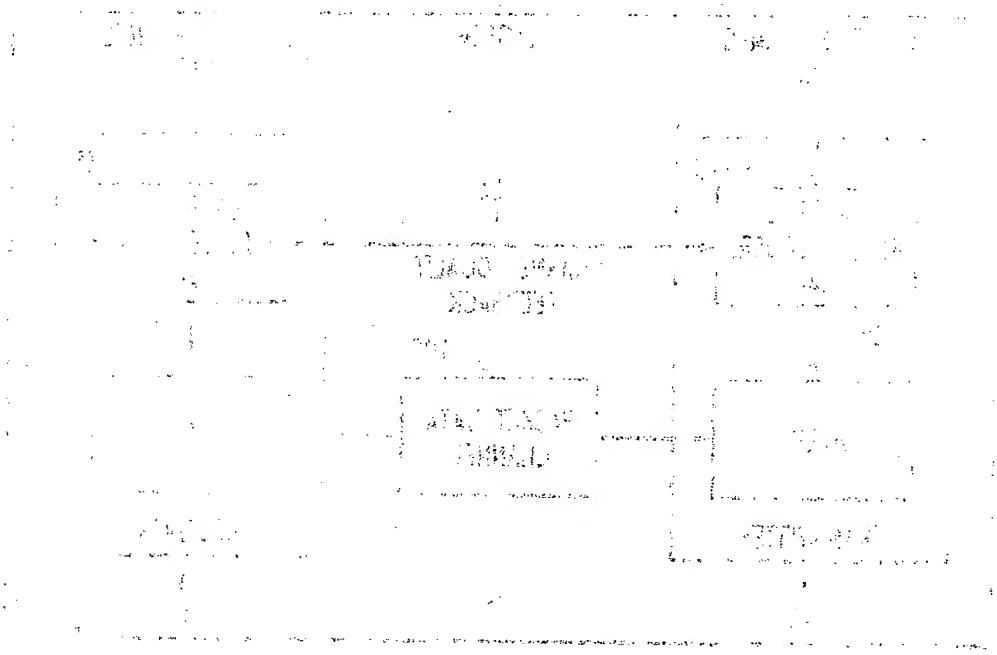
36. The method of claim 35 wherein the header includes a header type identifier and a data block signal sequence number for selected parity block signals.

BNSDOCID: <EP___0938207A2_I_>

37. The method of claim 35 including the steps of storing in a memory information corresponding to the N data bits and N parity bits, transmitting the D data block signals corresponding to the radio link protocol, decoding the Bi frame, and transmitting additional P parity block signals corresponding to the radio link protocol when whenever the Bi frame is not correctly decoded.

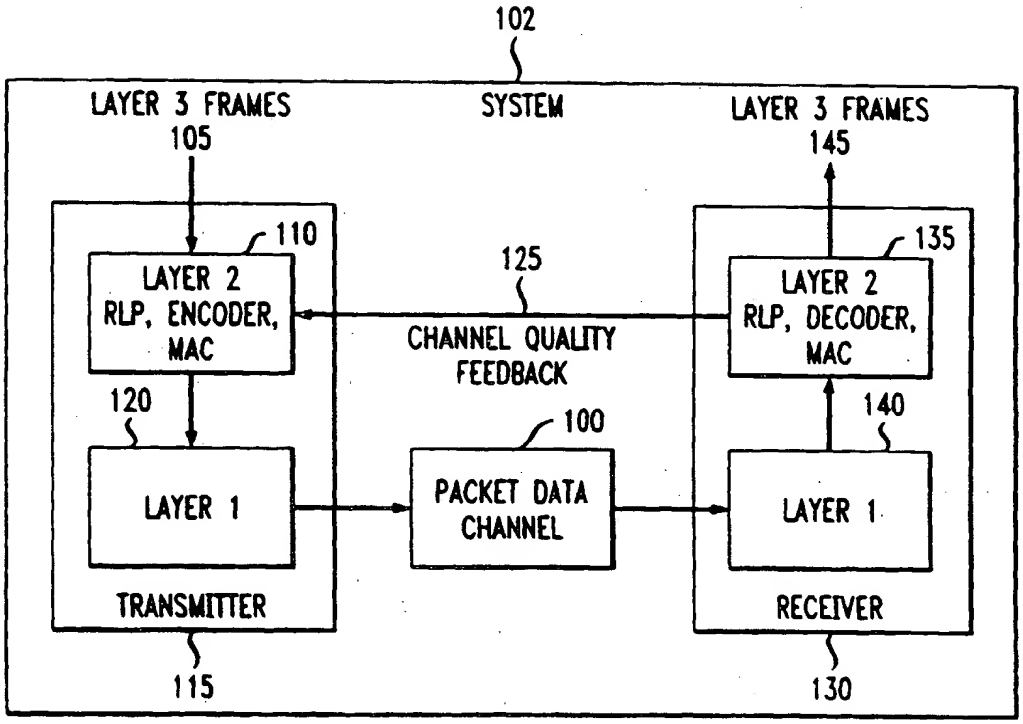
38. The method of claim 37 including the step of clearing the memory whenever the Bi frame is correctly decoded.

39. The method of claim 37 in which the memory includes a bitmap for delivered D data block signals and P parity block signals.



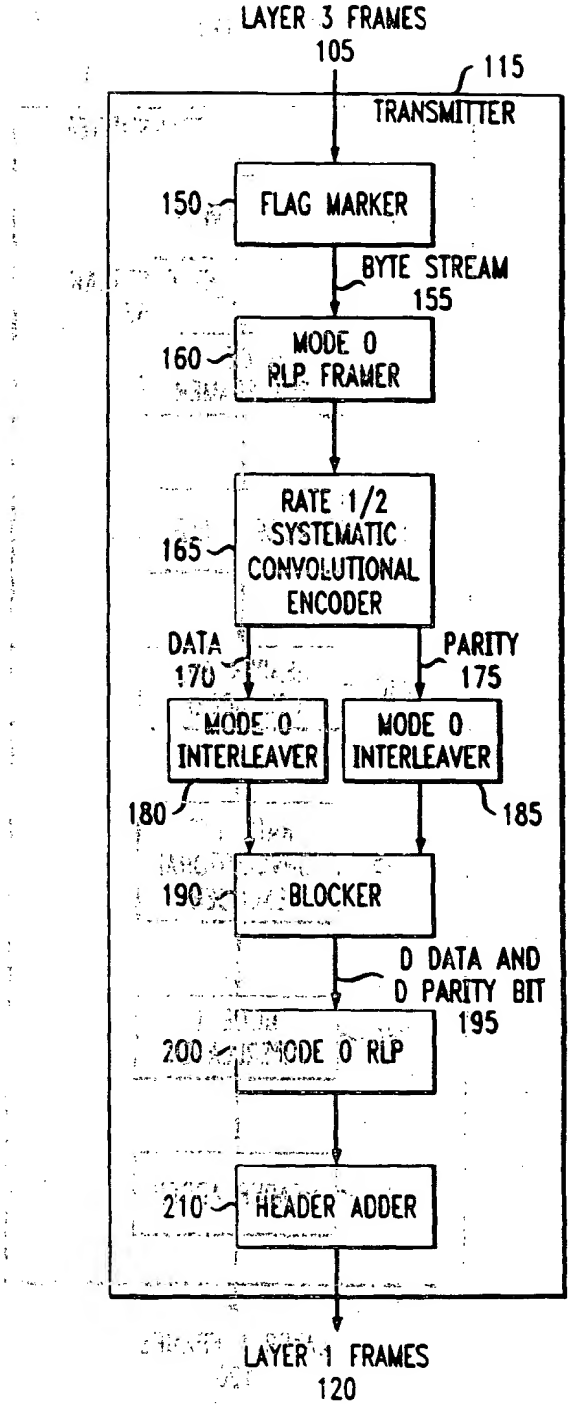
BNSDOCID: <EP___0938207A2_I_>

FIG. 1



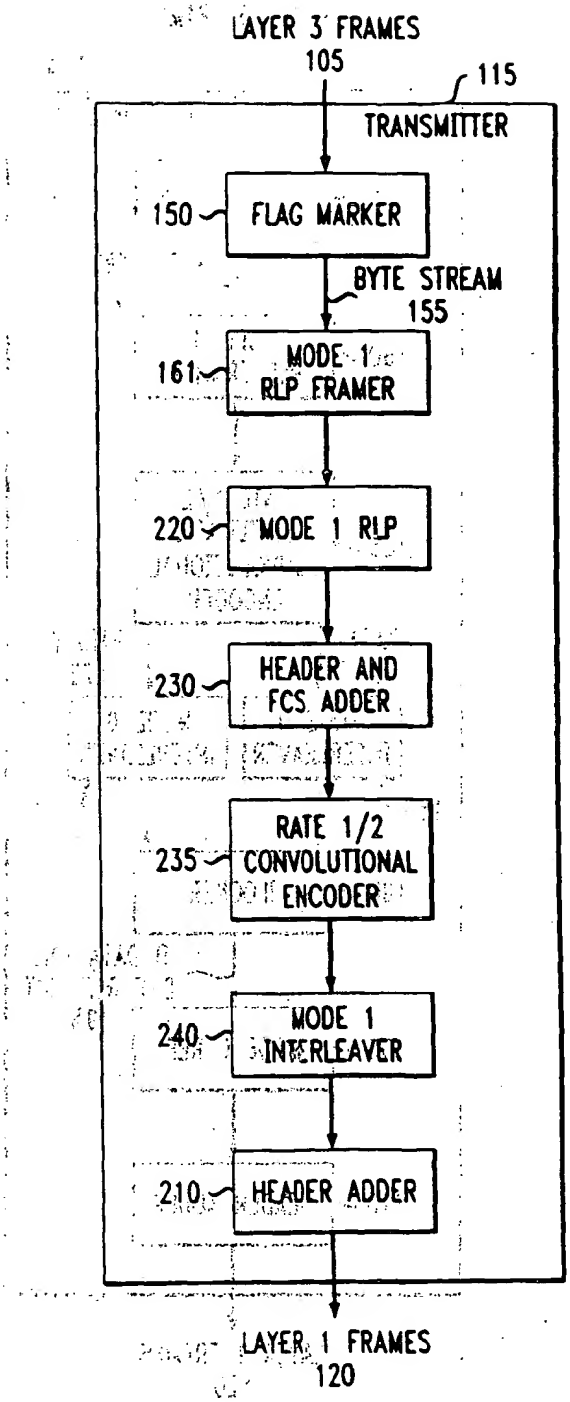
BNSDOCID: <EP___0938207A2_I_>

FIG. 2



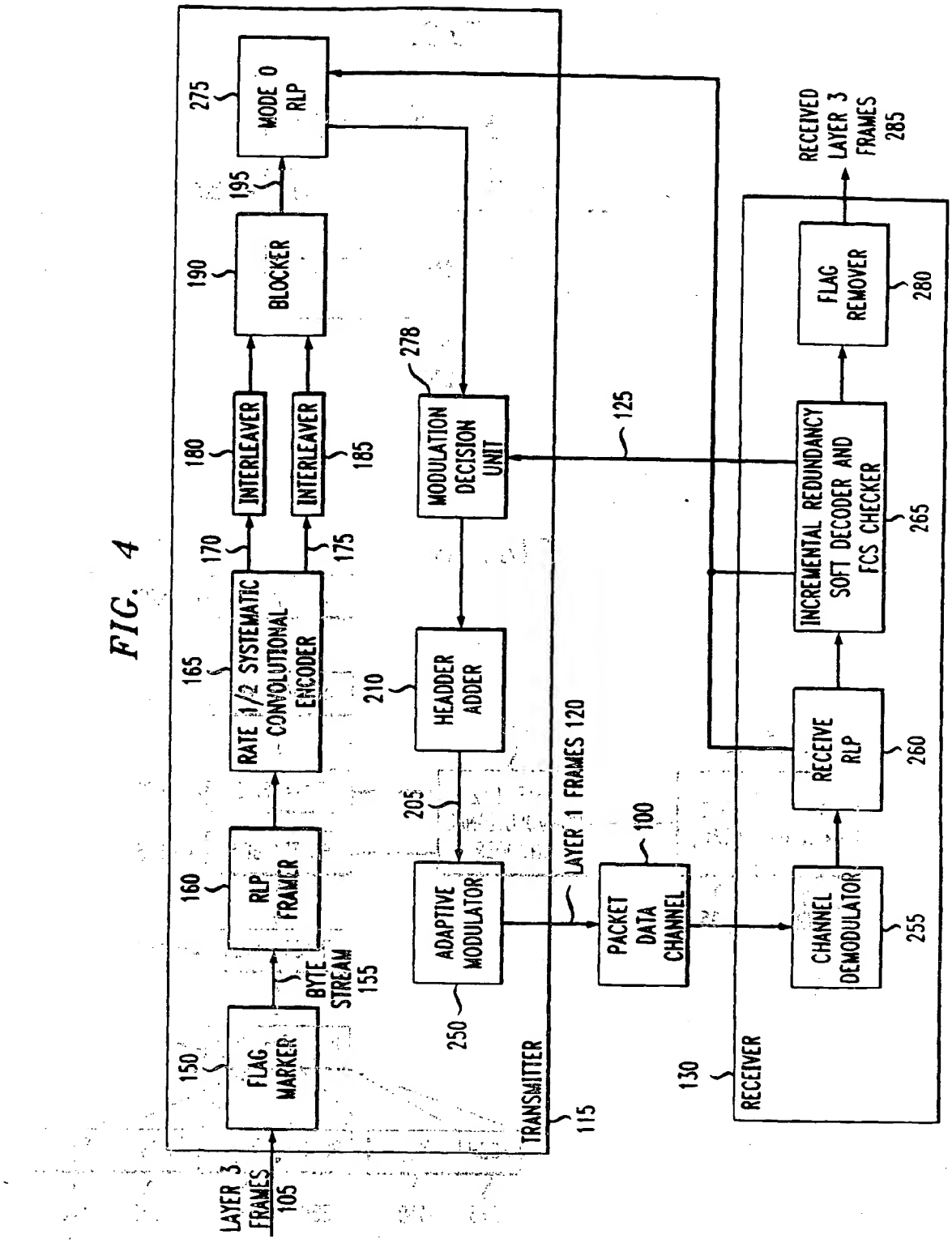
BNSDOCID: <EP___0938207A2_I_>

FIG. 3



BNSDOCID: <EP___0938207A2_I_>

FIG. 4



BNSDOCID: <EP___0938207A2_I_>

FIG. 5

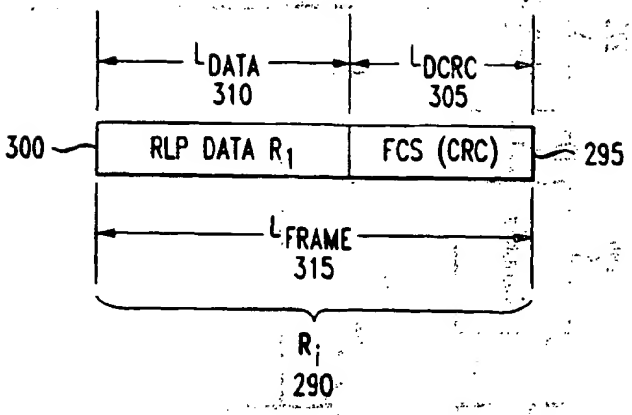
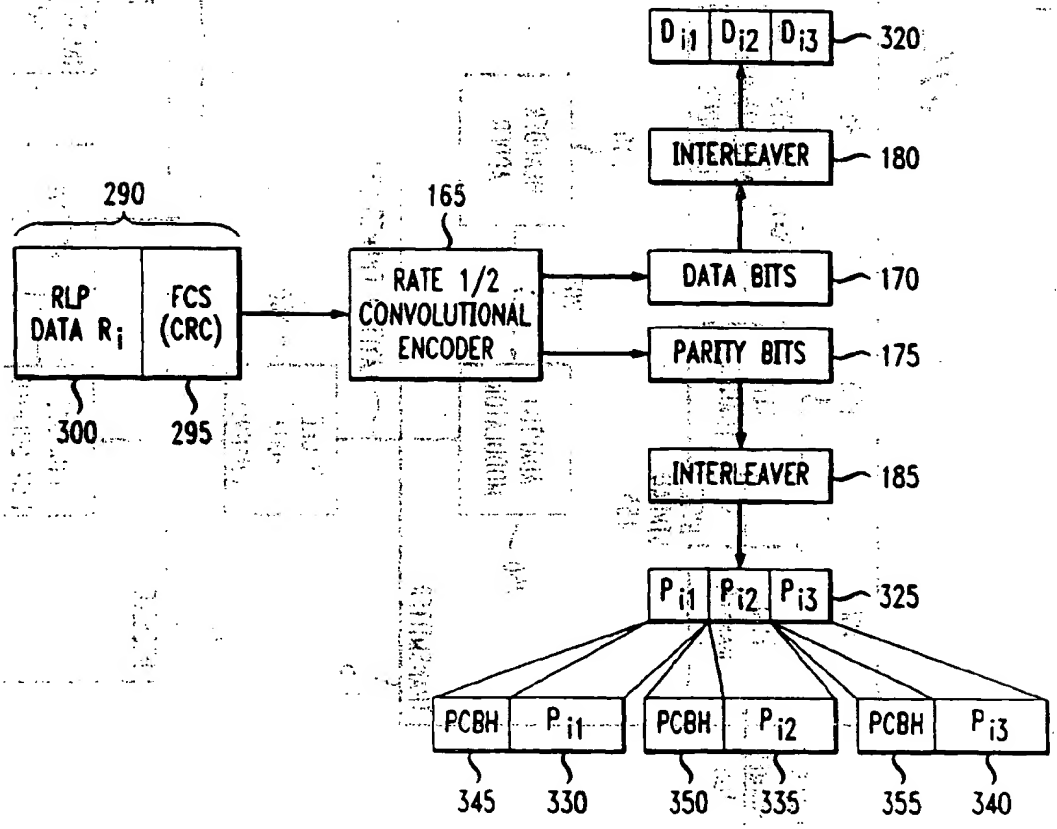


FIG. 6



BNSDOCID: <EP___0938207A2_I_>

FIG. 7

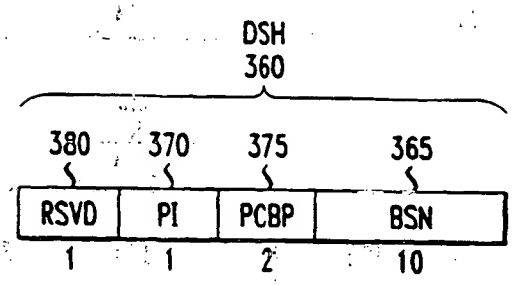
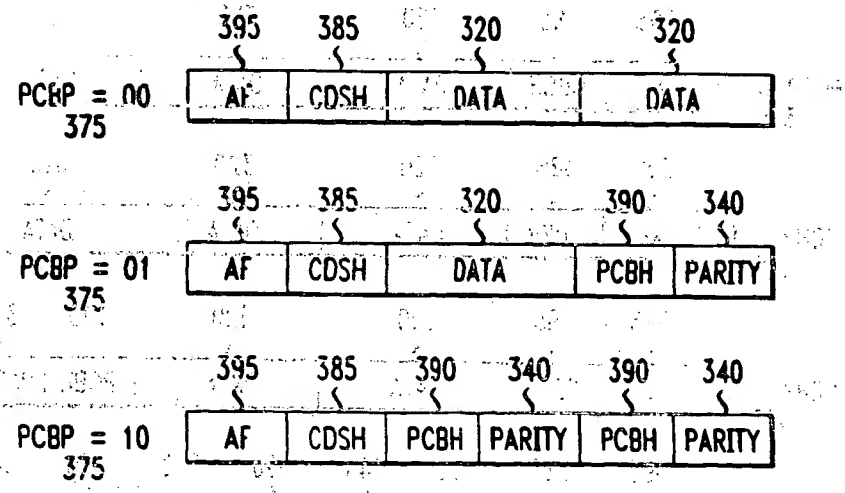


FIG. 8



BNSDOCID: <EP___0938207A2_I_>

FIG. 9

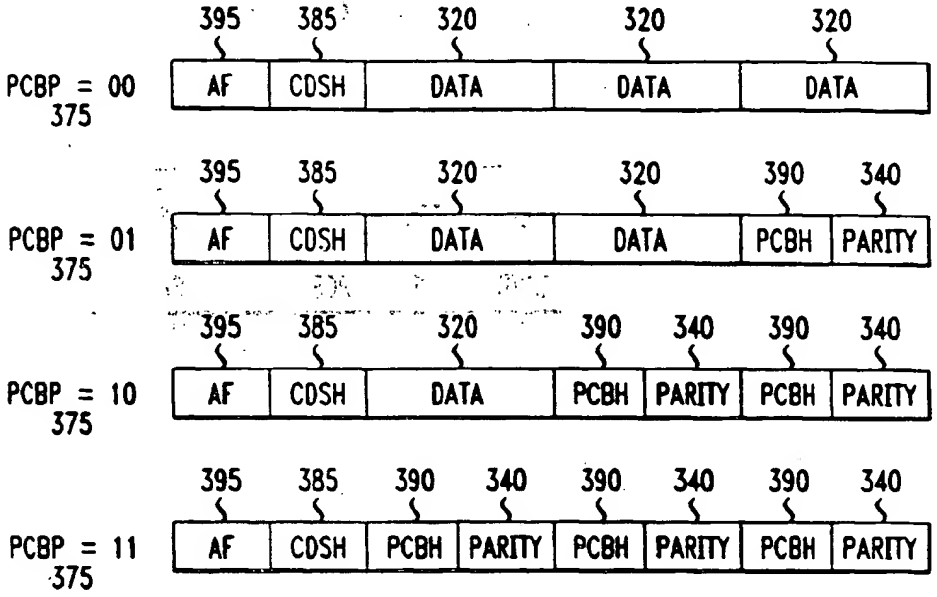
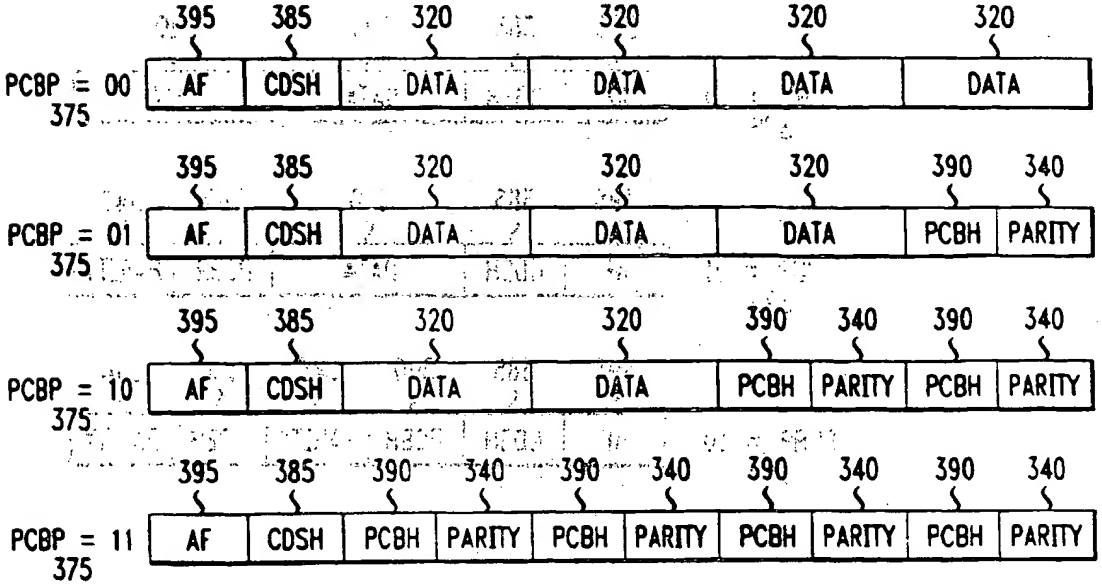
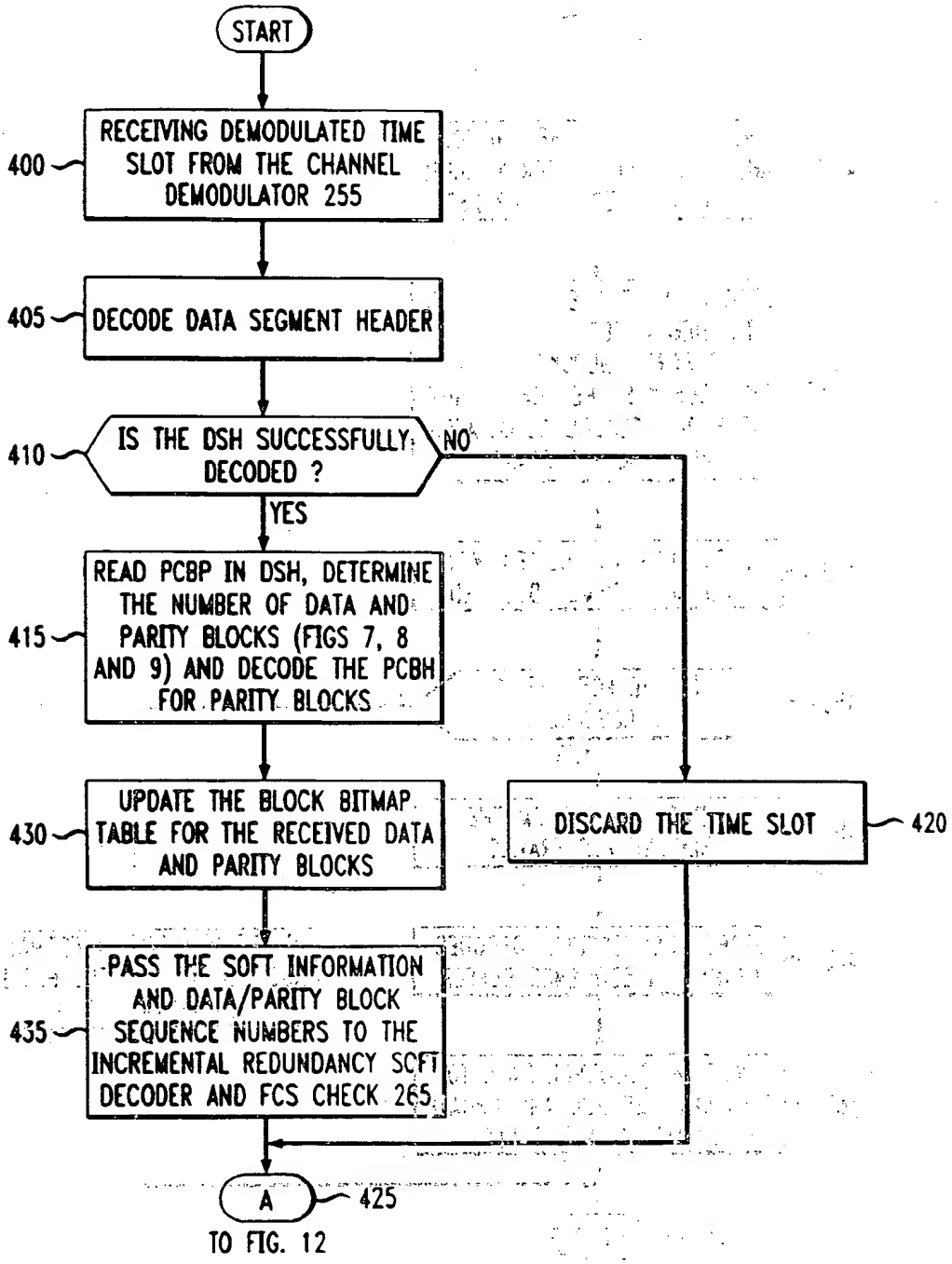


FIG. 10



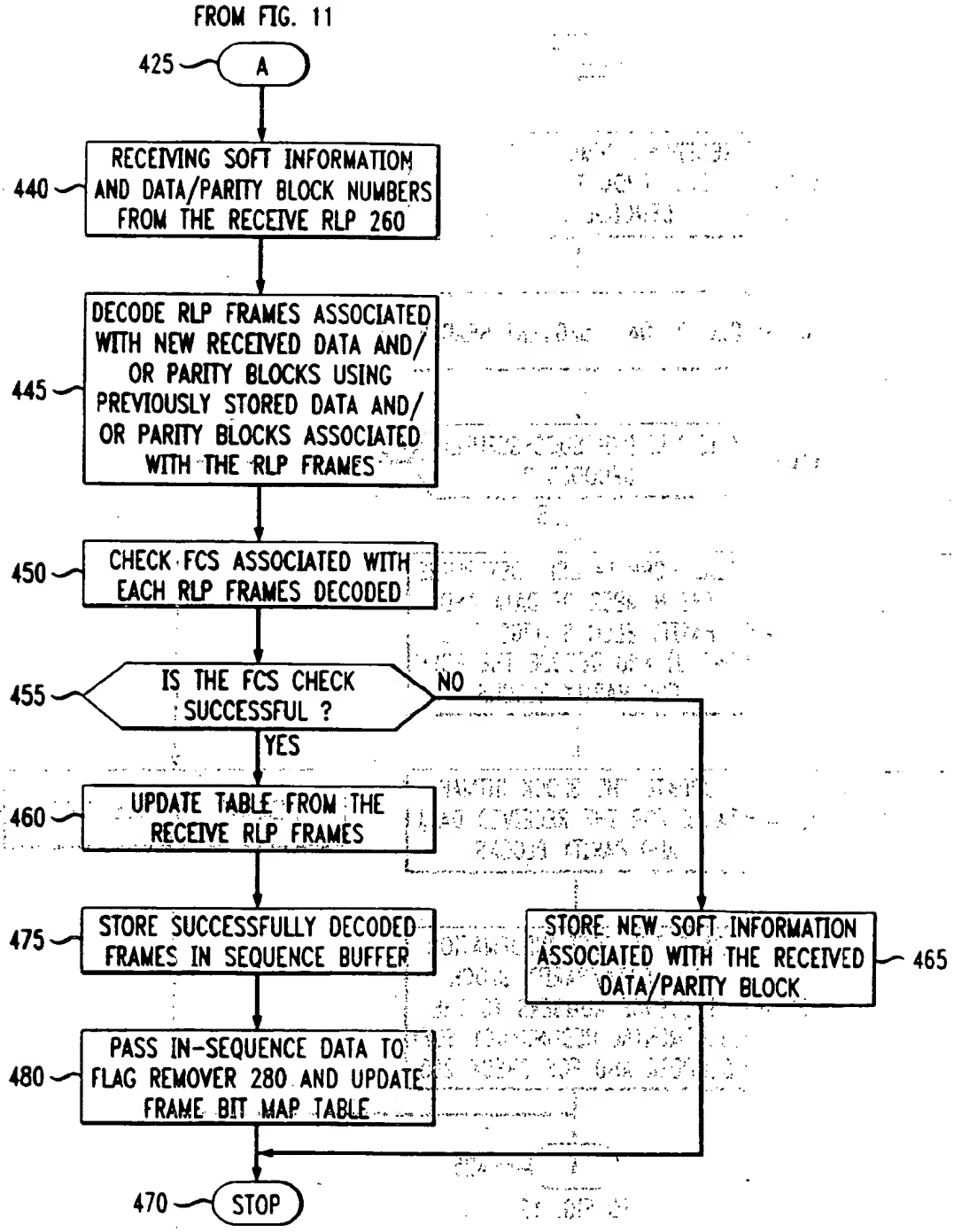
BNSDOCID: <EP___0938207A2_I_>

FIG. 11



BNSDOCID: <EP___0938207A2_I_>

FIG. 12



BNSDOCID: <EP___0938207A2_I_>

FIG. 13

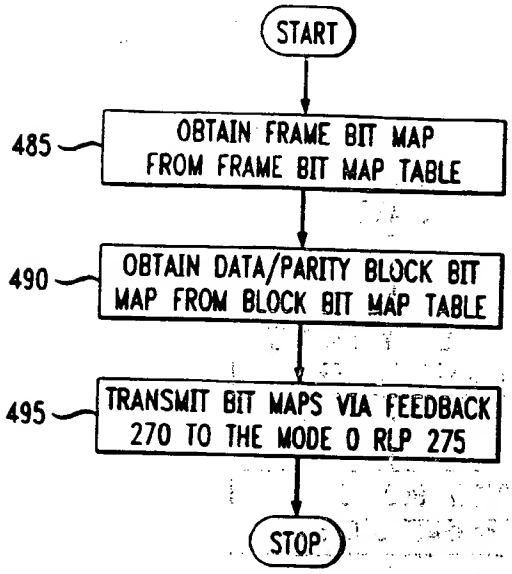
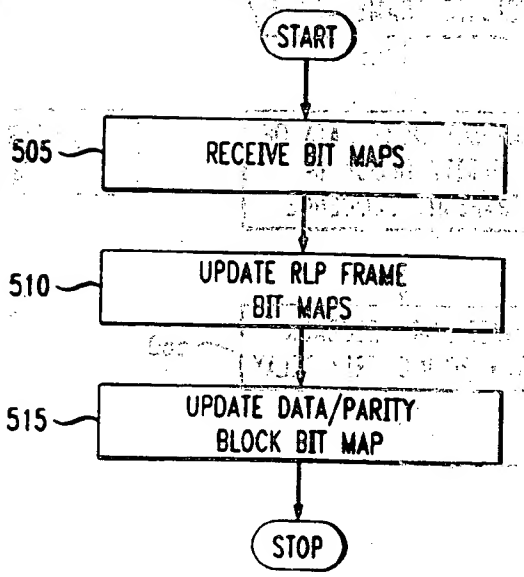
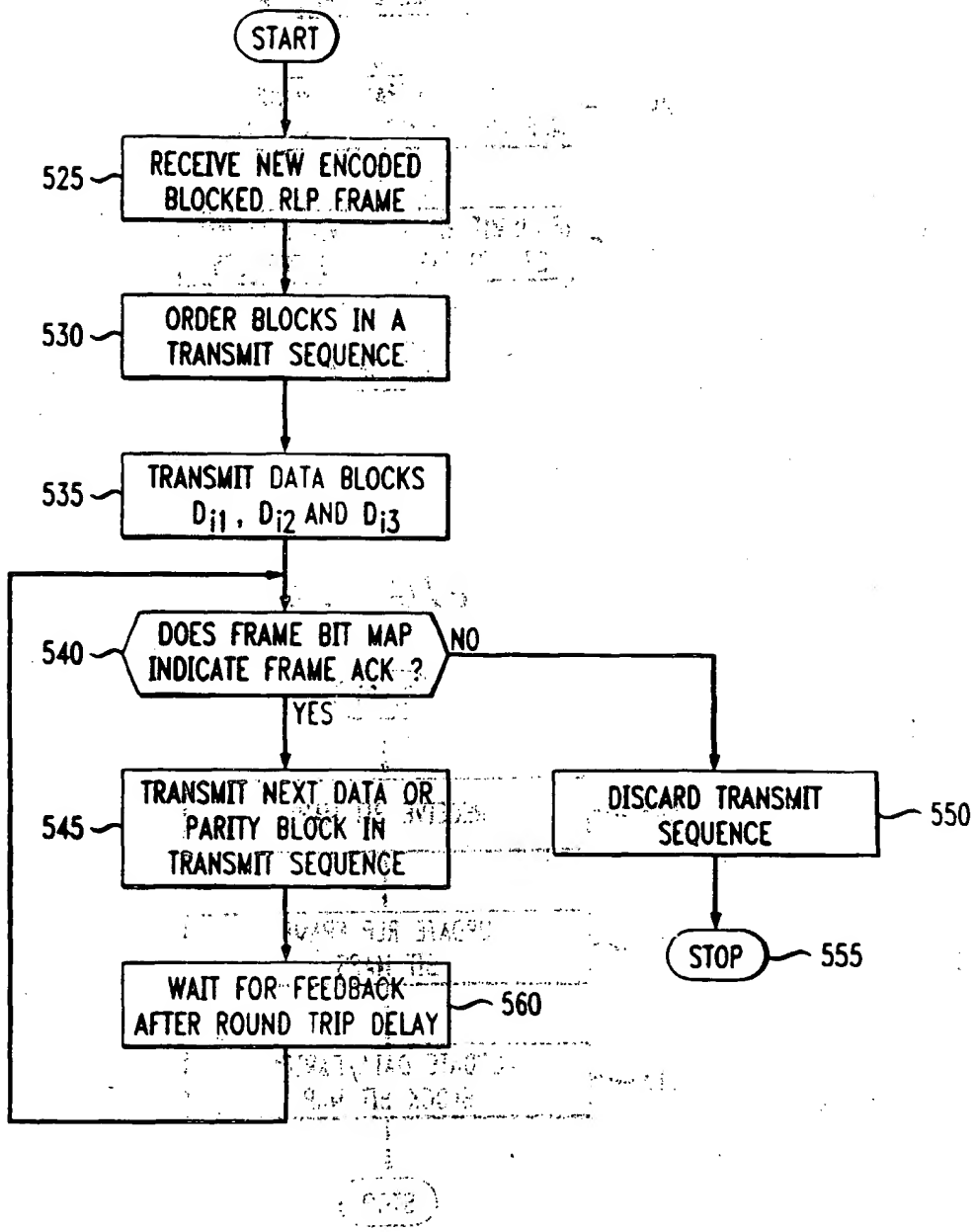


FIG. 14



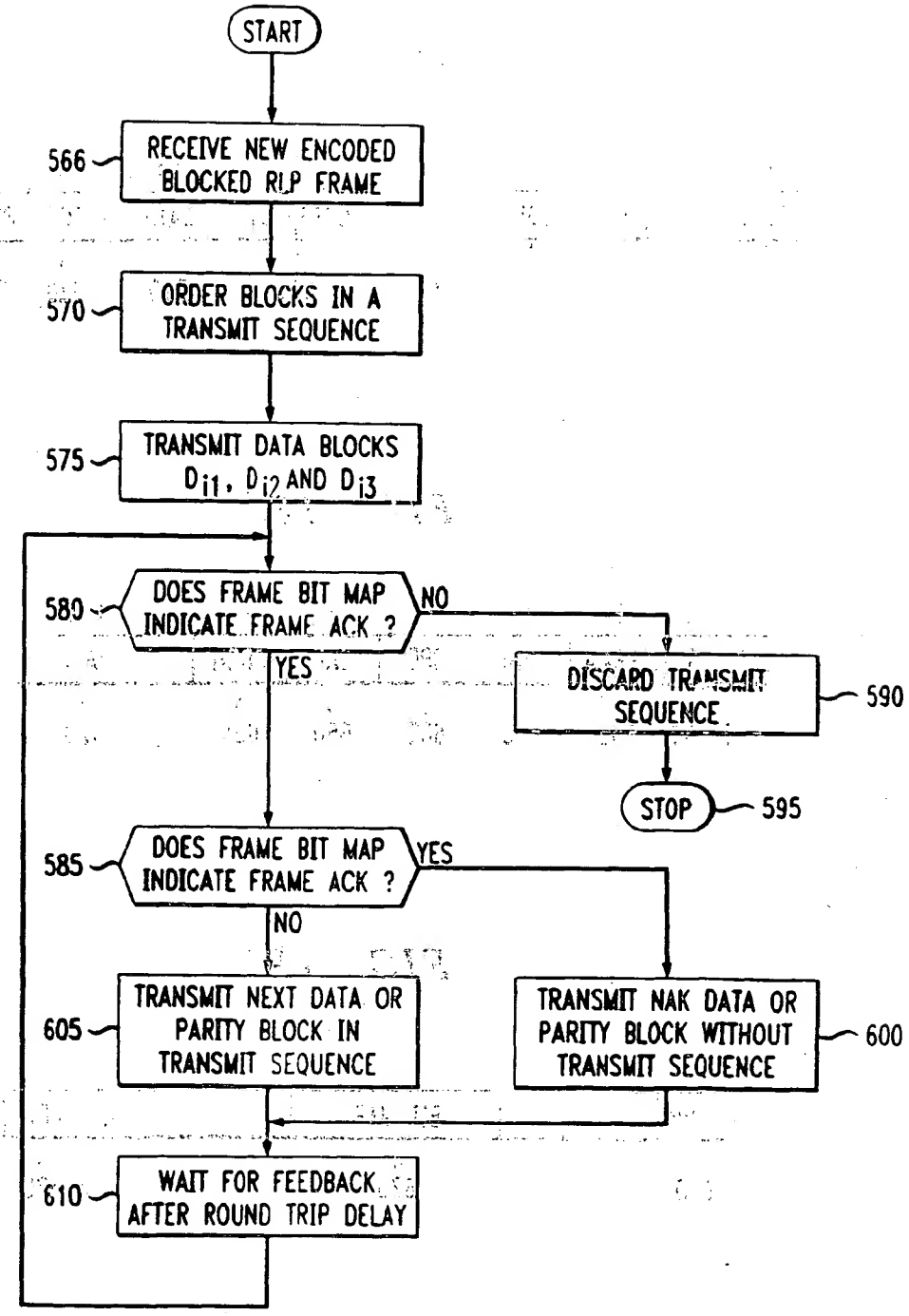
BNSDOCID: <EP___0938207A2_I_>

FIG. 15



BNSDOCID: <EP___0938207A2_I_>

FIG. 16



BNSDOCID: <EP___0938207A2_I_>

FIG. 17

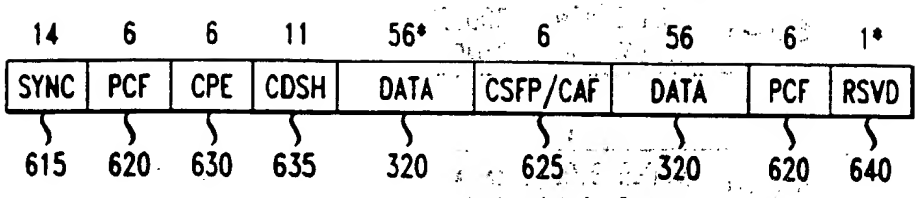


FIG. 18

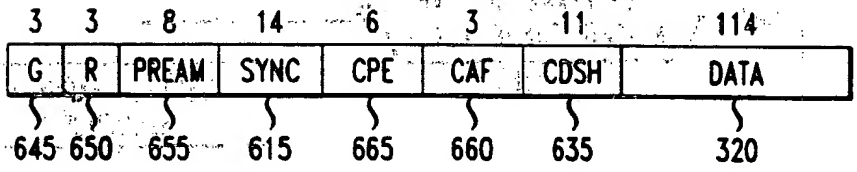
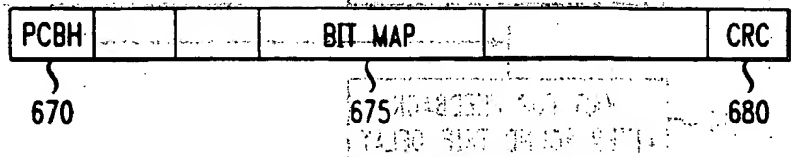
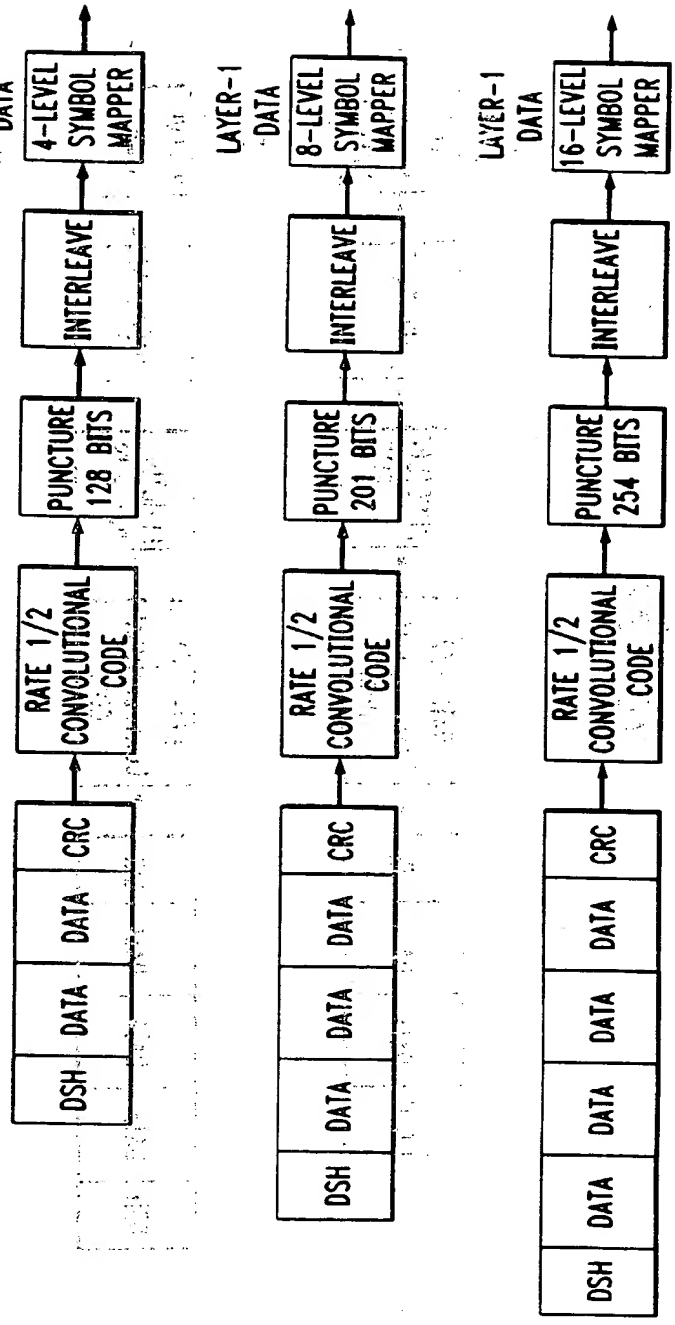


FIG. 19



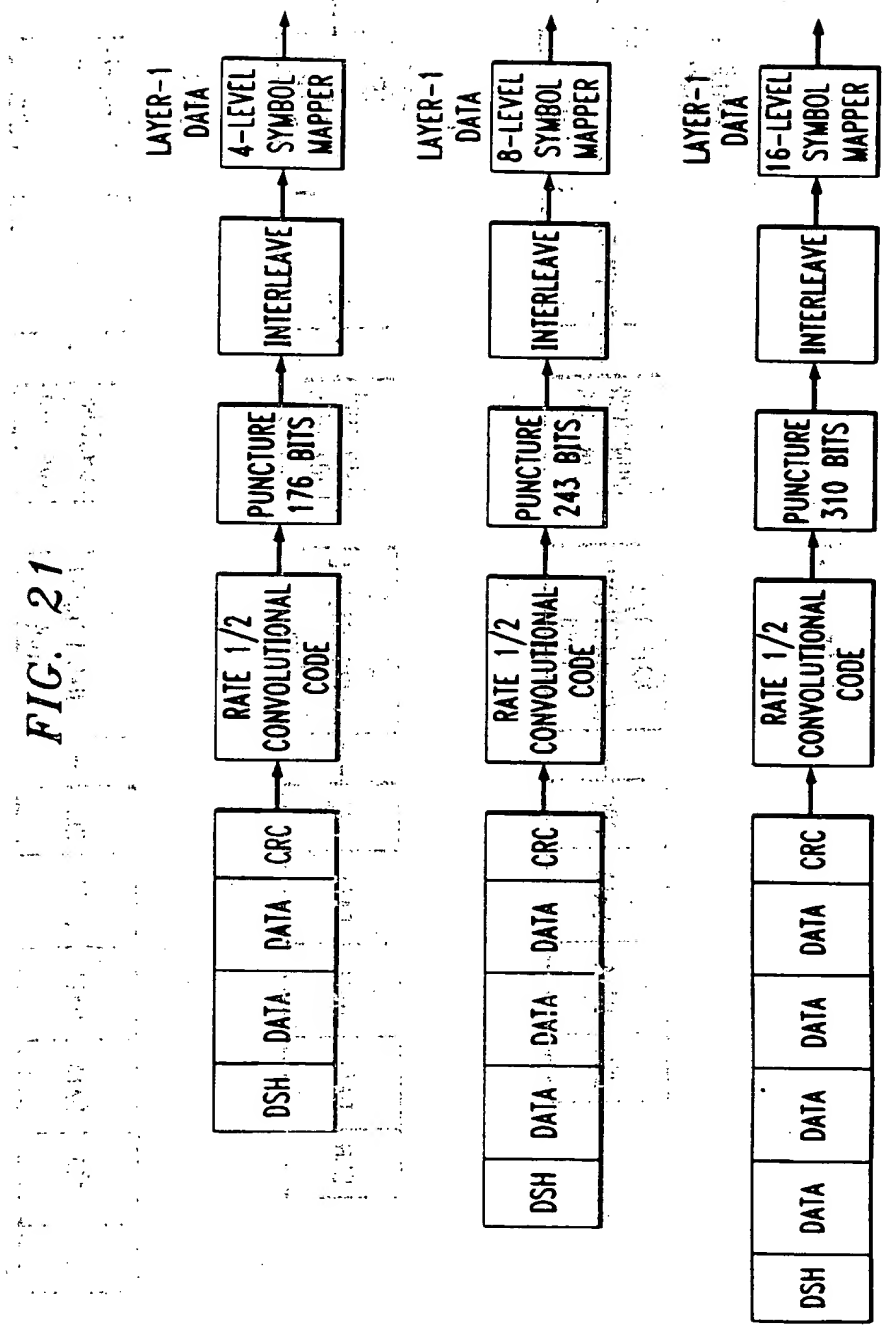
BNSDOCID: <EP___0938207A2_I_>

FIG. 20



BNSDOCID: <EP___0938207A2_I_>

FIG. 21



BNSDOCID: <EP___0938207A2_I_>

FIG. 22

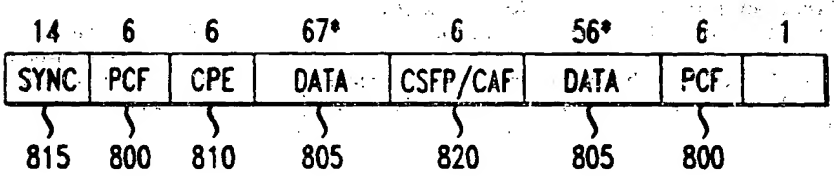
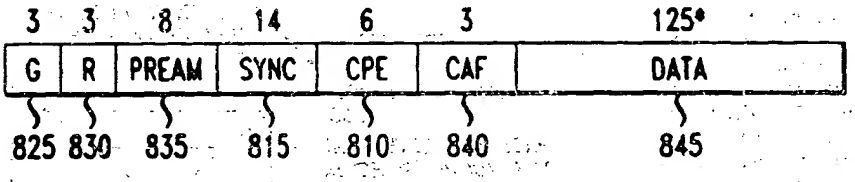


FIG. 23



BNSDOCID: <EP___0938207A2_I_>